View metadata, citation and similar papers at core.ac.uk
brought to you by CORE
provided by Caltech Authors

Submitted to *Interfaces*
manuscript (Please, provide the manuscript number!)

Authors are encouraged to submit new papers to INFORMS journals by means of a style file template, which includes the journal title. However, use of a template does not certify that the paper has been accepted for publication in the named journal. INFORMS journal templates are for the exclusive purpose of submitting to an INFORMS journal and should not be used to distribute the papers in print or online or to submit the papers to another publication.

Optimal Design of Efficient Rooftop Photovoltaic Arrays

Madeleine Udell
Cornell University, Ithaca, New York,udell@cornell.edu

Oliver Toole
Aurora Solar, San Francisco, California, otoole@aurorasolar.com

This paper addresses a major challenge in the residential solar industry: automated design of cost-effective, efficient rooftop photovoltaic (PV) installations. Optimal designs choose system components, locations, and wiring to minimize cost while meeting desired energy output and complying with all physical and legal constraints. We present a novel lower bound for the energy produced by a PV installation, which admits efficient optimization via integer linear programming. The resulting algorithm can design systems with a variety of solar hardware, including microinverters, string inverters, and DC optimizers, and optimize for complex shading patterns. Prior to our work, solar installers designed PV installations by hand. Our algorithm automates PV design using OR techniques, and has been used to create more than 70,000 designs for PV installations.

We compare the performance of our optimal designs to designs produced by solar installation experts at the National Renewable Energy Laboratory. Our algorithm designs faster, cheaper, more energy-efficient installations than expert installers, producing designs in tens of seconds where experts require tens of minutes. The optimized designs deliver the required energy output at lower cost in more than 70% of cases, and on average increase the energy produced per dollar invested. These results indicate that rooftop solar PV installations could produce 2% more energy at the same installation cost, or 820 GWh more energy per year.

This supplement clarifies the mathematical details of the problem formulations proposed in the main text, and describes a few approximations used to speed up the solution of the problems. We also formally show the hardness of the PV design problem.

Key words: Photovoltaic array, solar energy, solar system design, mixed-integer linear programming

History: This paper has been refereed.

Detailed Problem Formulation

Notation. To avoid confusion, we use a mnemonic notation for indexing panels, inverters, roof faces, strings, and time. Panels are indexed by $p \in \mathcal{P}$, inverters are indexed by $i \in \mathcal{I}$, roof faces are indexed by $f \in \mathcal{F}$, strings are indexed by $s \in \mathcal{S}$, and time is indexed by $t \in \mathcal{T}$. Feasible string lengths for inverter $i \in \mathcal{I}$ are indexed by $l \in \mathcal{L}_i$. The feasible string lengths will generally be a range of integers of the form $\mathcal{L}_i = \{m_i, \ldots, M_i\}$ where $m_i \in \mathbf{Z}$ and $M_i \in \mathbf{Z}$ are determined by the minimum and maximum allowable voltage for inverter i. The set of all allowable string lengths in the installation $\mathcal{L}$ satisfies

$$\mathcal{L} = \cup_{i \in \mathcal{I}} \mathcal{L}_i.$$

The set of panels on face f is denoted by $\mathcal{P}_f$; the sets $\mathcal{P}_f$ form a partition of $\mathcal{P}$.

We will use these indices to differentiate among variables and parameters corresponding to different system components. For example, e_{pt} will be the energy of panel p at time t, while e_{st} will be the energy of string s at time t.

Problem data. The problem data consists of the following:

- costs of panels c_p for $p \in \mathcal{P}$, the set of potential panels
- costs of inverters c_i for $i \in \mathcal{I}$, the set of potential inverters
- annual energy of panels e_p for $p \in \mathcal{P}$
- hourly energy of panels e_{pt} for $p \in \mathcal{P}, t \in \mathcal{T}$ (usually $\mathcal{T}$ ranges over every hour in a year)
- desired system energy $\bar{E}^{\text{des}}$
- constraints on inverter power, current, and voltage $\mathcal{C}_i$ for $i \in \mathcal{I}$
- maximum number of strings that can be wired to each inverter N_i for $i \in \mathcal{I}$

Bisection Algorithm

The overall scheme for the algorithm used in the Aurora AutoDesigner is described in Algorithm 1.

Algorithm 1 Bisection algorithm for PV array optimization

Input: Roof specification, desired energy E^{des}, convergence tolerance ϵ.

Output: Strings of panels, inverter wiring.

1: Choose potential panels and inverters.

2: Initialize $\bar{E}^{\text{des}} \leftarrow E^{\text{des}}$, $\bar{E}^{\text{high}} \leftarrow E^{\text{des}}$, $\bar{E}^{\text{low}} \leftarrow 0$.

3: **while** $\bar{E}^{\text{high}} - \bar{E}^{\text{low}} \geq \epsilon$ **do**

4: Solve design MILP to determine PV array configuration.

5: Simulate design to determine true energy output $\hat{E}$ of design.

6: Bisect: $\bar{E}^{\text{des}} \leftarrow \frac{1}{2}(\bar{E}^{\text{low}} + \bar{E}^{\text{high}})$. If $\hat{E} \geq E^{\text{des}}$, $\bar{E}^{\text{high}} \leftarrow \bar{E}^{\text{des}}$. Otherwise, $\bar{E}^{\text{low}} \leftarrow \bar{E}^{\text{des}}$.

7: Refine design with local search.

When the energy interval is small enough, all MILP target energies in the interval will produce the same design as one of the two end points of the interval. For example, if the set of panels $\mathcal{P}^{\text{low}}$ is used in the design that produces energy $\bar{E}^{\text{low}}$ and panel p is the next best panel, then we often eventually find that the following equality holds:

$$\bar{E}^{\text{low}} + e_p = \bar{E}^{\text{high}},$$

where e_p is the annual energy produced by the next best panel p.

Design MILP

We will now proceed to describe the design MILP in full mathematical detail.

Variables. We introduce integer variables to denote which components are used and how they are wired together:

- $z_p \in \{0,1\}$ for $p \in \mathcal{P}$ is 1 if panel p is used
- $z_i \in \{0,1\}$ for $i \in \mathcal{I}$ is 1 if inverter i is used
- $z_s \in \{0,1\}$ for $s \in \mathcal{S}$ is 1 if string s is used
- $z_{sp} \in \{0,1\}$ for $s \in \mathcal{S}$, $p \in \mathcal{P}$ is 1 if panel p is assigned to string s
- $z_{li} \in \{0,1\}$ for $i \in \mathcal{I}$, $l \in \mathcal{L}_i$ is 1 if any string of length l is wired to inverter i
- $n_{li} \in \mathbb{Z}$ for $i \in \mathcal{I}$, $l \in \mathcal{L}_i$ denotes the number of strings of length l wired to inverter i
- $n_{lf} \in \mathbb{Z}$ for $f \in \mathcal{F}$, $l \in \mathcal{L}$ denotes the number of strings of length l on face f

We introduce continuous variables to denote the energy produced by the components:

- $e_{st} \in \mathbb{R}$ for $s \in \mathcal{S}$, $t \in \mathcal{T}$ denotes the energy produced by string s at time t

Potential inverters. We pick inverters to use by solving a simple knapsack problem. For each type of inverter i, let the variable n_i be the number of inverters of type i, and recall that m_i is the minimum length of a string connected to an inverter of type i. Let n_p be a lower bound on the number of panels needed to meet the desired energy constraint $\bar{E}^{\text{des}}$. We can find a set of inverters needed to service n_p panels by solving the following optimization problem with variables $n_i \in \mathbb{Z}$:

$$\begin{array}{lll} \text{minimize} & \sum_{i\in\mathcal{I}} c_i n_i & \text{(cost)} \\ \text{subject to} & n_p \le \sum_{i\in\mathcal{I}} n_i m_i & \text{(capacity).} \end{array} \tag{1}$$

Capacity constraint. Define the length of each (potential) string, l_s for $s \in \mathcal{S}$. To ensure that these variables have the correct interpretations, we say the capacity constraint

$$(z_p, z_i, z_s, z_{li}, z_{sp}, n_{li}, n_{lf}) \in \mathcal{C}^{\text{cap}}$$

holds if

$$\begin{array}{lll}
\sum_{s\in\mathcal{S}} z_s = \sum_{f\in\mathcal{F}} n_{lf}, & l\in\mathcal{L} & \text{(every string is on some face)} \\
\sum_{s\in\mathcal{S}} z_s = \sum_{i\in\mathcal{I}} n_{li}, & l\in\mathcal{L} & \text{(every string is wired to some inverter)} \\
\sum_{s\in\mathcal{S}} z_{sp} = z_p, & p\in\mathcal{P} & \text{(no panel is on more than one string)} \\
\sum_{p\in\mathcal{P}} z_{sp} = l_s z_s, & s\in\mathcal{S} & \text{(strings have correct lengths)} \\
\sum_{l\in\mathcal{L}_i} z_{li} \le z_i, & i\in\mathcal{I} & \text{(only selected inverters have strings)} \\
n_{li} \le N_i z_{li}, & i\in\mathcal{I}, l\in\mathcal{L}_i & \text{(indicator true if number is positive).}
\end{array} \tag{2}$$

Note that the capacity constraint is representable as an affine inequality constraint in the problem variables.

Inverter constraint. The maximum current, voltage, and power across an inverter are linear in the problem variables, with constants of proportionality that depend on the characteristics of the type of panel chosen.

- Current is proportional to $\sum_{l\in\mathcal{L}_i} n_{li}$, the number of strings wired to the inverter.
- Voltage is proportional to $\max_{l\in\mathcal{L}_i} l z_{li}$, the maximum length of any string.
- Power is proportional to $\sum_{l\in\mathcal{L}_i} l n_{li}$, the total number of panels wired to the inverter.

For ease of notation, we assume that the constants of proportionality have been incorporated into the inverter constraint sets $\mathcal{C}_i$, so

$$\left(\sum_{l\in\mathcal{L}_i} n_{li},\ \max_{l\in\mathcal{L}_i} l z_{li},\ \sum_{l\in\mathcal{L}_i} l n_{li}\right) \in \mathcal{C}_i \tag{3}$$

if and only if the configuration is within the safe operating range for inverter i.

Energy. We use a simple approximation to the energy output of an array based on the irradiance and wiring to encourage similarly shaded panels to be strung together into strings. Letting $\mathcal{P}_s$ denote the set of panels in string s, and e_{pt} the energy of panel p and time t, the approximation to the energy e_{st} produced by string s at time t is

$$e_{st} = |\mathcal{P}_s| \min_{p\in\mathcal{P}_s} e_{pt}.$$

This approximation is an underestimate of the true energy. The MPPT will choose a current through the string which is feasible (less than the maximum current allowable for any panel in the string), and which causes the string to produce the greatest energy possible. The current that produces the maximal power in the panel with the least irradiance is always feasible for the other panels. It produces power $\min_{p\in\mathcal{P}_s} e_{pt}$ in the panel with the least irradiance, and produces at least that much power in every other panel in the string. Figure 2b in the main text demonstrates this approximation graphically.

This approximation has the advantage that it can be represented with linear constraints, using a big-M relaxation. If for some $M \in \mathbf{R}$,

$$e_{st} \leq e_{pt} l_s + M(1 - z_{sp}), \quad s \in \mathcal{S}, t \in \mathcal{T}, p \in \mathcal{P}, \tag{4}$$

then e_{st} is a lower bound on the energy produced in string s at time t.

- If $z_{sp} = 1$, then $e_{st} \leq e_{pt} l_s$.
- If $z_{sp} = 0$, then $e_{st} \leq e_{pt} l_s + M$.

If M is sufficiently large, then e_{st} is not constrained by e_{pt} for panels p not chosen to be in string s. Choosing

$$M \geq \left(\max_{l\in\mathcal{L}} l\right)\left(\max_{p\in\mathcal{P}, t\in\mathcal{T}} e_{pt}\right)$$

is sufficient.

This linear approximation is exact when, at each time t, every panel in a given string has exactly the same energy. In fact, this happens surprisingly often, since at any time of day, the energies of all the panels on a given roof face can be well approximated by one of two values: $e_{pt} \approx \alpha_t$ or $e_{pt} \approx \beta_t$ for every $p \in \mathcal{P}$. This simplification is due to the binary impact of shading: panels either produce a high energy (when in direct sunlight) or a low energy (using diffuse light from the blue sky). Using this linear approximation in the optimization problem induces a clustering of panels into strings that groups panels by the energy they produce. One nice result is that the resulting clustering frequently groups shaded panels together and groups unshaded panels together, making the approximation exact at the solution for many times t.

Design MILP. Now we are ready to put together all of the parts into the design MILP. Our problem is

$$\begin{array}{llll}
\text{minimize} & \sum_{i\in\mathcal{I}} c_i z_i + \sum_{p\in\mathcal{P}} c_p z_p & & \text{(cost)} \\
\text{subject to} & E^{\text{des}} \leq \sum_{s\in\mathcal{S}} \sum_{t\in\mathcal{T}} e_{st} & & \text{(energy)} \\
& e_{st} \leq e_{pt} l_s + M(1 - z_{sp}) & s\in\mathcal{S}, t\in\mathcal{T}, p\in\mathcal{P} & \text{(linear approximation (4))} \\
& e_{st} \leq M z_s & s\in\mathcal{S} & \text{(no energy from unused string)} \\
& (\sum_{l\in\mathcal{L}_i} z_{li}, \max_{l\in\mathcal{L}_i} l z_{li}, \sum_{l\in\mathcal{L}_i} l n_{li}) \in \mathcal{C}_i & i\in\mathcal{I} & \text{(inverters (3))} \\
& (z_p, z_i, z_s, z_{li}, z_{sp}, n_{li}, n_{lf}) \in \mathcal{C}^{\text{cap}} & & \text{(capacity (2))}
\end{array} \tag{5}$$

with variables $z_p \in \{0,1\}$, $z_i \in \{0,1\}$, $z_s \in \{0,1\}$, $z_{li} \in \{0,1\}$, $z_{sp} \in \{0,1\}$, $n_{li} \in \mathbb{Z}$, and $n_{lf} \in \mathbb{Z}$, for $p \in \mathcal{P}$, $i \in \mathcal{I}$, $f \in \mathcal{F}$, $s \in \mathcal{S}$, and $l \in \mathcal{L}$.

The design MILP gives the minimum-cost set of panels z_p and inverters z_i together with an assignment of panels to strings z_{sp} and of strings to faces n_{lf} with the properties that (1) the panels can be wired safely to inverters; and (2) the linear approximation to the energy produced by the strings of panels exceeds the desired energy $\bar{E}^{\text{des}}$.

Improving Performance

The design MILP (5) produces a design that satisfies the requirements of the PV design problem. However, the size of the problem is too large to solve in a reasonable time for all but the most simple problems. In this section, we describe a few approximations that reduce the time needed to produce a good solution to the PV design problem to less than a minute.

Assignment and Wiring

One important change to the design MILP to reduce computation time is to split it into two separate optimization problems, which we call the assignment MILP and the wiring MILP. The assignment MILP is independent of the stringing of the panels, while the wiring MILP performs the stringing.

Capacity constraints. The capacity constraint for the assignment problem now takes a different form: it enforces constraints on the string lengths, but no longer enforces any constraints on the assignment of panels to strings (which we leave to the wiring MILP). To ensure that the remaining variables have the correct interpretations, we say that the capacity constraint for the assignment problem

$$(z_p, z_i, z_{li}, n_{li}, n_{lf}) \in \mathcal{C}^{\text{asg}}$$

holds if

$$\begin{array}{lll}
\sum_{p \in \mathcal{P}_f} z_p = \sum_{l \in \mathcal{L}} l n_{lf}, & f \in \mathcal{F} & \text{(every panel is on some face)} \\
\sum_{l \in \mathcal{L}_i} n_{li} \le N_i z_i, & i \in \mathcal{I} & \text{(every string is wired to some inverter)} \\
\sum_{i \in \mathcal{I}} n_{li} = \sum_{f \in \mathcal{F}} n_{lf}, & l \in \mathcal{L} & \text{(every string is on some face).}
\end{array} \tag{6}$$

Note that the capacity constraint for the assignment problem is representable as an affine inequality constraint in the problem variables.

Assignment MILP. The assignment MILP assigns panels to roof faces and to inverters, respecting the capacity constraints by solving the problem

$$\begin{array}{lll}
\text{minimize} & \sum_{i \in \mathcal{I}} c_i z_i + \sum_{p \in \mathcal{P}} c_p z_p & \text{(cost)} \\
\text{subject to} & \sum_{p \in \mathcal{P}} e_p z_p \ge E^{\text{des}} & \text{(energy)} \\
& (\sum_{l \in \mathcal{L}_i} z_{li}, \max_{l \in \mathcal{L}_i} l z_{li}, \sum_{l \in \mathcal{L}_i} l n_{li}) \in \mathcal{C}_i \; i \in \mathcal{I} & \text{(inverters (3))} \\
& (z_p, z_i, z_{li}, n_{li}, n_{lf}) \in \mathcal{C}^{\text{asg}} & \text{(capacity (6))}
\end{array} \tag{7}$$

with variables $z_p \in \{0,1\}$, $z_i \in \{0,1\}$, $z_{li} \in \{0,1\}$, $n_{li} \in \mathbb{Z}$, and $n_{lf} \in \mathbb{Z}$ for $p \in \mathcal{P}$, $i \in \mathcal{I}$, $f \in \mathcal{F}$, and $l \in \mathcal{L}$. The solution to this problem gives the minimum cost set of panels and inverters with the properties that

1. the panels can be assigned to faces and wired safely to inverters, and
2. the sum of the annual energies of the panels exceeds the desired energy E^{des}.

This problem has fewer variables than the design MILP (5), and can be solved much more efficiently.

Wiring MILP. The solution to (7) determines the number of strings of each length on each face. Given these strings $s \in \mathcal{S}$ each with length l_s, we solve the following problem for each face to find the best way to wire the panels together:

$$\begin{array}{llll} \text{maximize} & \sum_{s\in\mathcal{S}} \sum_{t\in\mathcal{T}} e_{st} & & \text{(energy)} \\ \text{subject to} & e_{st} \le e_{pt} l_s + M(1 - z_{sp}) & s \in \mathcal{S}, t \in \mathcal{T}, p \in \mathcal{P} & \text{(linear approximation (4))} \\ & \sum_{p\in\mathcal{P}} z_{sp} = l_s & s \in \mathcal{S} & \text{(string length)} \\ & \sum_{s\in\mathcal{S}} z_{sp} \le 1 & p \in \mathcal{P} & \text{(one string per panel)} \end{array} \tag{8}$$

with variables $z_{sp} \in \{0,1\}$ and e_{st}.

Binary Representation

We can exploit other properties of our problem data to achieve a more refined solution. As we noted earlier, at any given time, panels on the same roof face either produce a high energy (when in direct sunlight) or a low energy (using diffuse light from the blue sky). Hence, the energy of the panels on a particular roof face can be well approximated as

$$e_{pt} \approx \alpha_t (1 - w_{pt}) + \beta_t w_{pt},$$

where $w_{pt} \in \{0,1\}$, and α_t (β_t) is the average energy of an unshaded (shaded) panel at time t. The binary constraints on the problem variables allows ILP solvers to restrict the search space significantly, leading to faster convergence.

Clustering Times

The complexity of Problem (8) grows significantly with $|\mathcal{T}|$. To make the problem smaller, we can cluster times to find a subset of times that still captures the shading information. The clustering method described here builds on De Rubira and Toole (2015).

Recall that we defined $e_t \in \mathbf{R}^{N_p}$ to be the vector of panel energies at time t. Suppose that we have a partition $\mathcal{T}_1, \ldots, \mathcal{T}_k$ of the times $\mathcal{T}$ so that e_t is identical for every $t \in \mathcal{T}_i$, for each $i = 1, \ldots, k$. Pick a set of index times $t_1 \in \mathcal{T}_1, \ldots, t_k \in \mathcal{T}_k$. Then Problem (8) reduces to

$$
\begin{array}{llll}
\text{maximize} & \sum_{s\in\mathcal{S}} \sum_{i=1}^{k} |\mathcal{T}_i| e_{st_i} & & \text{(energy)} \\
\text{subject to} & e_{st_i} \le e_{pt_i} l_s + M(1 - z_{sp}) & s \in \mathcal{S}, i = 1, \ldots, k, p \in \mathcal{P} & \text{(linear approximation (4))} \\
& \sum_{p\in\mathcal{P}} z_{sp} = l_s & s \in \mathcal{S} & \text{(string length)} \\
& \sum_{s\in\mathcal{S}} z_{sp} \le 1 & p \in \mathcal{P} & \text{(one string per panel).}
\end{array}
\tag{9}
$$

Problem (9) uses many fewer variables and constraints than Problem (**??**. Hence, this clustered problem formulation can typically be solved in less than a second, several orders of magnitude faster than the original formulation.

For more general problems, we may still wish to cluster times so that e_t is approximately equal for every time in the cluster. To achieve this, first pick the number of time clusters k. This number will be chosen to ensure Problem 9 can be solved efficiently. We aim to find sets $\mathcal{T}_1, \ldots, \mathcal{T}_k$ that partition $\mathcal{T}$, as well as representative times for each cluster $t_i \in \mathcal{T}_i$ for $i = 1, \ldots, k$. To do this, we solve

$$
\begin{array}{ll}
\text{minimize} & \sum_{i=1}^{k} \sum_{t\in\mathcal{T}_i} \|e_t - e_{t_i}\|, \\
\text{subject to} & \{\mathcal{T}_i\} \text{ partition the set } \mathcal{T} \\
& t_i \in \mathcal{T}_i, \quad i = 1, \ldots, k
\end{array}
$$

with variables $\mathcal{T}_i$ and t_i.

An alternative approach, which we employ in our numerical experiments, is to approximate the energy vectors e_t in a cluster by the cluster centroid b_i. We solve

$$\begin{array}{ll} \text{minimize} & \sum_{i=1}^k \sum_{t\in\mathcal{T}_i} \|e_t - b_i\| \\ \text{subject to} & \{\mathcal{T}_i\}_{i=1}^k \text{ partition the set } \mathcal{T} \\ & b_i \in \{0,1\}^{|\mathcal{P}|}, \quad i = 1,\ldots,k. \end{array}$$

This problem is a standard k-means clustering problem. Note that the cluster centroid vector of energies b_i may never be (exactly) achieved at any time.

Shading patterns are periodic with only slight variations daily. Hence, a good choice of the number of clusters k can greatly reduce the problem size without introducing significant errors.

If we use the binary representation introduced above, then instead of clustering energies $e_t \in \mathbf{R}^{|\mathcal{P}|}$, we can instead cluster the boolean vectors $w_t \in \{0,1\}^{|\mathcal{P}|}$. Given this clustering, we solve Problem 9 with the objective (energy) replaced by

$$\sum_{s\in\mathcal{S}} \sum_{i=1}^k |\mathcal{T}_i| e_{st_i}.$$

Eliminate islands. Often times the design that produces the most energy is too expensive to install, or aesthetically awkward. One common problem is that panels may be placed far away from any other panels. We call these *islanded panels*. These islanded panels increase the cost of the installation, since a separate *rack* (beams attached to the roof) must be installed for each islanded panel. They also reduce the visual symmetry of the PV array, and so may be considered an eyesore.

To remove islanded modules from the design produced by the design MILP, we perform a local search around the given design by iteratively moving islanded panels to available locations with the most adjacent panels, so long as the move does not reduce the energy of the installation unacceptably. Let n_l be the number of adjacent panels to location l, and e_l be the annual energy of a panel at location l. The desirability of location l is given by the function

$$f(l) = n_l + \epsilon e_l,$$

where $\epsilon < 1/\max_l e_l$ trades off between our annoyance at including islanded modules and our annoyance at reducing the energy of the installation. We iteratively select the filled location l with the smallest value of $f(l)$, and move that panel to the unfilled location l' with the largest value of $f(l)$, until no unfilled location has a higher value of f than any filled location.

Hardness

Here we show the hardness of stringing solar panels by reduction from vertex cover. This proof follows an argument proposed by Schulman (2016). Here we show that an algorithm to string N solar panels into K strings to produce at least N units of power (according to our linear under-approximation to power production) over T time periods, with the constraint that every panel must produce energy, can be used to solve the vertex cover problem, which is known to be NP hard (Karp 1972).

The reduction is from vertex cover. Consider a graph with N edges and T vertices. To each vertex, we associate a time. To each edge between vertices i and j, we associate a solar panel whose hourly energy production is given by the vector $e_i + e_j \in \{0,1\}^T$, the sum of the unit vectors in the i and jth directions. That is, the (i,j)th solar panel produces unit energy at time i and time j, and nothing at any other time. Using microinverters, each panel would produce two units of energy, and the whole array would produce $2n$ units of energy.

We show there is a vertex cover of size K if and only if there is a stringing of N panels into K strings so that every panel produces energy at at least one time (out of T).

Suppose there is a vertex cover of size K. Create one string corresponding to each vertex t in the cover, and place in that string every solar panel whose corresponding edge is covered by vertex t, breaking ties arbitrarily. We see that every string produces power at the time t corresponding to the vertex, and so every panel produces power. Hence, the total energy produced is at least N.

Conversely, suppose we have a grouping of panels into K strings so that every panel produces power. Because every panel in a string produces power at the same time, for each string we may pick a time t when all panels in the string produce power. (Indeed, if there is more than one panel in the string, the string produces power at only one time, because no two edges in the graph span the same pair of vertices.) The vertex corresponding to time t covers the edges corresponding to the panels in the string. Picking one such vertex for each string gives a K vertex cover for the graph.

Acknowledgments

This work was developed with support from Aurora Solar and the DOE Sunshot Initiative. Aurora Solar also provided access to its shading simulation, which computes the objective function for the AutoDesigner.

References

De Rubira TT, Toole O (2015) Clustering techniques for photovoltaic system simulations under nonuniform irradiance conditions. *2015 IEEE Photovoltaic Specialist Conference (PVSC)*, 1–5 (Institute of Electrical and Electronics Engineers, Washington, DC).

Karp RM (1972) Reducibility among combinatorial problems. Miller RE, Thatcher JW, Bohlinger JD, eds., *Complexity of computer computations*, 85–103 (Springer, Boston).

Schulman L (2016) Email to Madeleine Udell, March 10.